FSA Integration Partner
United States Department of Education
Federal Student Aid

eZ-Audit
CONFIGURATION MANAGEMENT PLAN

Task Order #116

Version 3.0

Final

10/31/2003

Version History

Version	Date	Description	By Whom
1	1/20/2003	Initial Creation	Maja Dragnic
2	4/13/2003	Update with new team members	Andre Sakaluk
3	8/15/2003	Update per SSO comments.	Matt Portolese
4	8/20/2003	Added additional detail for environments	

Table of Contents

1.0 Configuration Management Plan Overview

1.1 Introduction

Configuration Management (CM) enables the controlled and repeatable management of information technology (IT) architecture components as they evolve in all stages of development and maintenance. CM implements a process by which the project teams and stakeholders identify, communicate, implement, document and manage changes in the systems environment. When properly implemented, CM ensures the integrity of the items that have been placed under its control.

1.2 Purpose

The purpose of this CM Plan is to establish a sound CM approach that maintains the integrity of eZ-Audit systems and provides traceability for changes incorporated into the environment. The CM process integrates the technical and administrative actions of identifying the functional, performance and physical characteristics of a configuration item (CI) and controls the changes to those characteristics.

2.0 Project Overview

Please refer to the Project Plan. The document defines and describes the project to which this CM plan applies. The document includes the scope of the project, timetable, milestones, key deliverables, roles and responsibilities, reporting structure, etc. The document defines and describes the project organizational chart to which this CM plan refers.

3.0 Configuration Management Approach

3.1 Configuration Management Overview

On the eZ-Audit team, CM is the means through which the integrity of the design, development, and other work products is maintained. CM is implemented through six processes: change control, baselining, archival, migrations, CM audit and reporting. This plan explains how eZ-Audit has designed and has planned to implement these processes to ensure the integrity of its systems and data. Further, this plan will outline the project's infrastructure and lay the framework that will be used to manage the implementation of CM at eZ-Audit.

3.2 Project Infrastructure

3.2.1 Development Environment

The following section outlines the Change Control process for tracking and managing changes.

Changes are inevitable; therefore managing them will require full integration and communication between the teams. Two types of changes have been defined:

- *Minor/Configuration changes* – Corrections or clarifications identified by either test or development team, no significant impact to schedule, does not require CCB or FSA approval

- *Major/Design changes* – Design or process changes identified by the test or development team, inability to fulfill requirements, missing data, results in an impact to schedule, requires CCB or FSA review

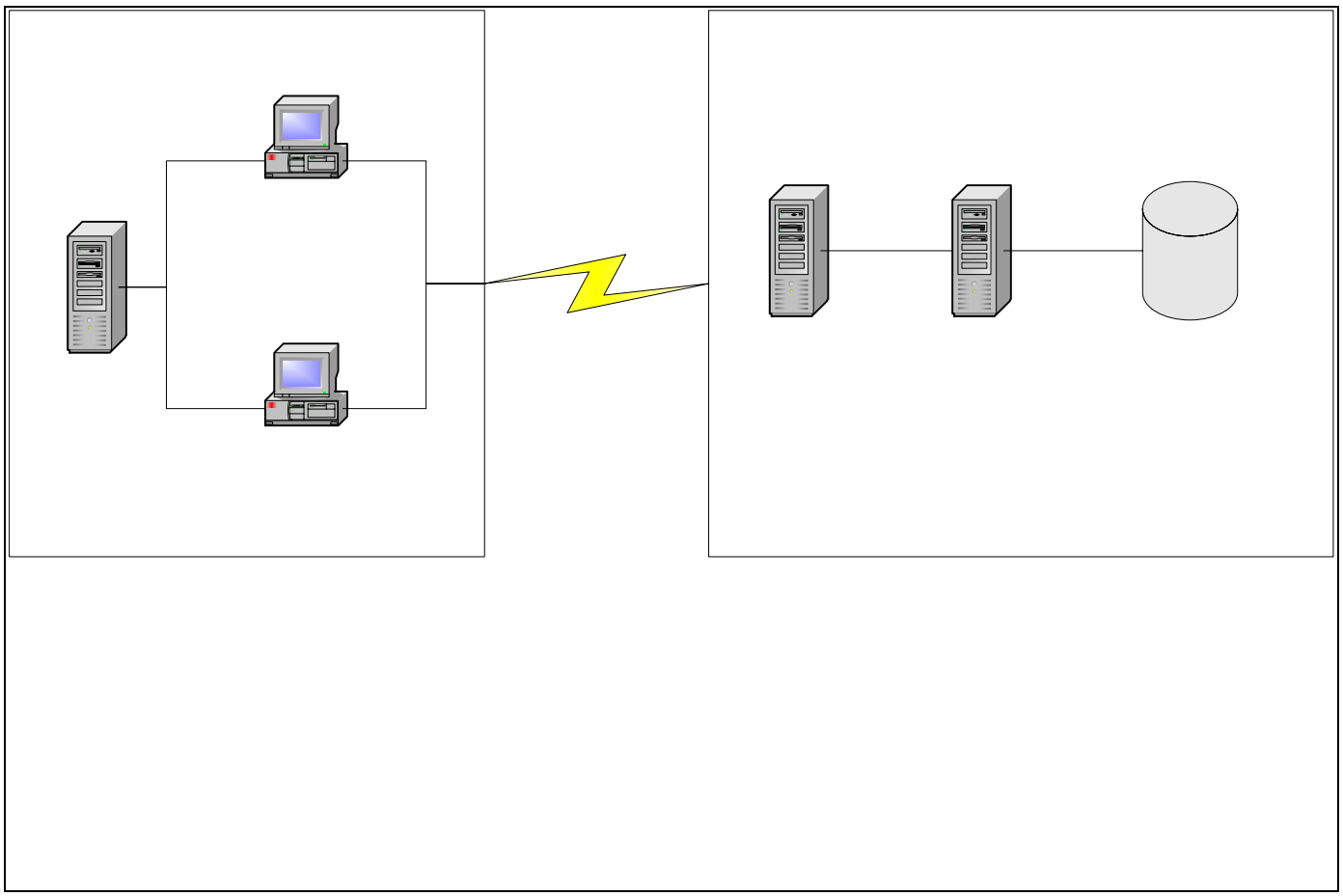
Seth Sinclair will serve as a functional expert and as a liaison for assessing Development team changes. They will assist with determining if a change is minor or requires FSA review. Seth will also serve as liaison for assessing Test Team changes. They will assist with determining if a functional change is minor or requires escalation to the Development team. The teams have established a Change Control process and tracking tool to monitor all changes.

3.2.1.1 Development Architecture

The eZ-Audit development environment is composed of three primary components:

1. **ITA Development Environment** – Includes a web server, application server, RCS components, and database instance for development and unit testing. The development servers are hosted at the VDC.
2. **Rational ClearCase** – Used for version control and source code management of application code and configuration modules.
3. **Development workstations** – Primary development machines for application and report development with connections to both the ITA Development Environment at the VDC and the Rational ClearCase code repository.

The diagram below illustrates the development environment components.



The development environment used on eZ-Audit is based on a standardized set of team tools.

Tool	Use
Visual Age for Java (VAJ)	Used primarily for the development of Java classes. VAJ also provides an integrated WebSphere Test Environment for local testing of all code.
Textpad (or other text editor)	Development of Java Server Pages (JSP) and database objects (stored functions, stored procedures, views, etc.)
Microstrategy Desktop	Development of Reports objects. All reports are based on database views.
Rational ClearCase	Repository for all code and build information.
Rational ClearQuest	Tracking tool for system incidence reports (SIR) and ECM requests
MS Access or Tookit for Oracle Application Developers (TOAD)	Used to view and manipulate data for testing efforts.

The eZ-Audit system is based on the Apache Struts framework and uses several reusable component services (RCS) from the ITA team. The most notable is the persistence framework, which allows for consistent database access methods. The email component is also used in eZ-Audit, both in the peps interface and for sending emails to users of the system. Documentation on both the RCS components and the typical path of a request in eZ-Audit can be found in the Application Architecture document.

3.2.1.2 Development Process

3.2.1.2.1 ClearCase

In order to properly track and maintain development information, the ClearCase repository must be configured. Depending on the current site of development, the ClearCase local client or the web client may be used. If possible, the local client is the recommended solution. It provides much more capability to the developer. For our purposes, we will describe the use of the web client, which was the only tool available for Release 1.01. The web client is a standard version control tool, used for checking in and out code. It is configured through a remote connection to the VDC and the local client on the host machine. Within ClearCase, all object are stored in a Virtual Object Branch or VOB. Items are located in a VOB and displayed within a View. One VOB may have many different views. A view is configured through its config spec. eZ-Audit has two branches on which development has occurred. Release 1.0 was developed on the main branch. Release 1.01 was developed on a new branch called r1_01. The view used in development today is called ezaudit_release101_view and pulls all of the latest Release 1.01 code and in the absence of something on the r1_01 branch, the latest version on the main branch is pulled. This will happen if the particular object was not updated in Release 1.01.

To view and modify the config spec, first open ClearCase Explorer on the host machine. Select the view you wish to modify on the left navigation bar, right click on it, and select properties. If the view you are modifying is not present, use the Base UCM tab to open the view. Once in the properties box, select the config spec tab and click the edit button to make any changes.

To create a new view, use the wizard on the Base UCM tab and follow the instructions. Create a dynamic view so that all changes will trickle down to the child views on the web client.
To use the web client, go to the http://www.fsatool.ed.gov website. From here, click on the ClearCase link. This will allow you to select from views you have already set up, or create a new view. Views in the web

client are slightly different than the local client. A snapshot view must be used in the web client. The information will be downloaded to your pc in a directory you specify. In order to get the latest versions, you will have to download all of the files you need. To create a new view, first name the view, then select the ezaudit_release101_view from the configuration follow list box and type in the directory you wish to use. The project standard directory is c:/clearcase/ezaudit101/. From here you may download all the classes and begin your work process.

To use the tool, simply navigate through the file structure on the web client and select the file you wish to check out. Click on the checkbox next to the file and click on the 'Checkout' button in the top navigation. Enter a comment when prompted. The file will be automatically downloaded to your computer in the directory you determined. After modifying the file, you may choose to check in, or undo the checkout of the file. If checking in, make sure the file is in the same location as when it was downloaded. Navigate the web client back to the file and select the checkbox next to the file. Use the 'Checkin' button or the 'Undo Checkout' button from the top navigation to complete your task.

If you have a new file to add to ClearCase, simply place the file in the desired directory on your computer. Select the 'Add' button from the top navigation. The web client will search for new files and prompt you to add them to ClearCase.

Labeling Code: Labeling code within ClearCase is an important step in creating baselines. To label code, you must use the Remote Connection to the host machine in the VDC. Select the Label Wizard from the Start > Programs > Rational ClearCase> menu. Following the instructions in the wizard and create a new label type.

Samson Abebe (Samson.abebe@ed.gov, 202-377-3532) is the FSA contact for Rational. He will be able to connect via RCO to the host machine for Rational (4.20.15.228) at the VDC.

3.2.1.2.2 Visual Age for Java

Detailed information regarding the setup and configuration of Visual Age for Java can be found in the Development Environment Configuration document. Working with VAJ and using the ClearCase web client is a simple process. However, failing to follow the correct procedure can lead to code replacement or overwrites. The first step is to checkout the files you wish to modify from ClearCase. If you are uncertain about a file, you may check it out now and undo the checkout later or wait to check it out, but do not edit a file without checking it out first. After checkout, import the class into Visual Age for Java. The simplest way to do this is to right click in the left navigation bar and select the import function. Import from a directory, and browse to file the file you wish to import. Now you may modify that file at will.

When completed, you may test your changes in the WebSphere Test Environment. Once again, detailed information may be located in the Development Environment Configuration document. If the changes are satisfactory, you may now export your code from VAJ back into the ClearCase directory, After exporting, use the web client to check your changes in.

3.2.1.2.3 Microstrategy Desktop

The Microstrategy desktop is the tool required to change reports. Detailed information on using the desktop can be found in the Microstrategy Reporting Documentation. All reports that are developed are based on views created in the database. Originally Microstrategy is designed to work with a star schema database (a database with information extracted from a transactional database and is configured specifically for reporting). Because we are using the same database for both reporting and our normal transactions, the

inherent SQL statements that the tool developed were causing the database to run out of temporary memory when running them due to the extremely high number of joins. In order to fix this problem, a view in Oracle was created for each report. Microstrategy then created each report based on its corresponding views and the views were able to be modified to improve performance. To modify or create a new report, begin by creating a view in the database to reference. Then follow the Microstrategy developers guide to complete the object development. All database objects must then be checked into ClearCase.

3.2.1.2.4 File Management Process

1) A Java file is checked out from the eZ-Audit ClearCase repository.
2) The developer imports the file into VAJ and makes the appropriate changes.
3) The local test environment is used to test the modified component.
4) The modified .java and the corresponding .class files are exported to the same directory it was initially imported from.
5) The modified .java file is checked back in to the ClearCase repository.
6) The .class file is FTP'd from the workstation to /www/dev/ezaudit/4 on su35e5 (4.20.15.135) for Assembly Testing. For Java class files, the WebSphere instance must be restarted for the change to take place. JSP files need no restart.
7) The file is executed and/or tested once on the development servers to insure coordination with the other classes. In the development mode, code may be moved in builds to coordinate the testing effort of many developers.

3.2.1.2.5 Oracle Connection

SERVICE_NAME: EZAUDEV
HOST: 4.20.15.15
PORT: 1691
Username: ezaudit
Password: audit12dev

3.2.1.2.6 Assembly Test Environment

The assembly test environment is used for developer testing of integrated components. This is especially important to test the ability of components to work together when developed by different people. To create a build, one person's desktop machine or a build machine must be designated. This will help ensure consistency with the build process.

1. Download all files to be moved to the environment to your local machine.
2. Import only the Java files into your Visual Age for Java environment.
3. Export the .class files into your local directory.
4. FTP all files to the /www/dev/ezaudit/4/ directory on su35e5
    a. All Java .class files are located in the lib folder
    b. All properties and configuration files are in the servlets folder
    c. All JSP files and struts config are located in the web folder.
    d. If any of the files are new, the permissions will have to be changed to 777
5. Restart WebSphere.
    a. Telnet into 4.20.25.135
    b. Login (you will need to apply for an id for this machine with ED)
    c. Execute the following command:

sudo /opt/dev35/WebSphere/AppServer/bin/restart_ezaudit.sh

6. All images, stylesheets, and javascript files must be loaded onto su35e2 in the /www/dev/ezaudit/htdocs directory.

3.2.2 Test Environment

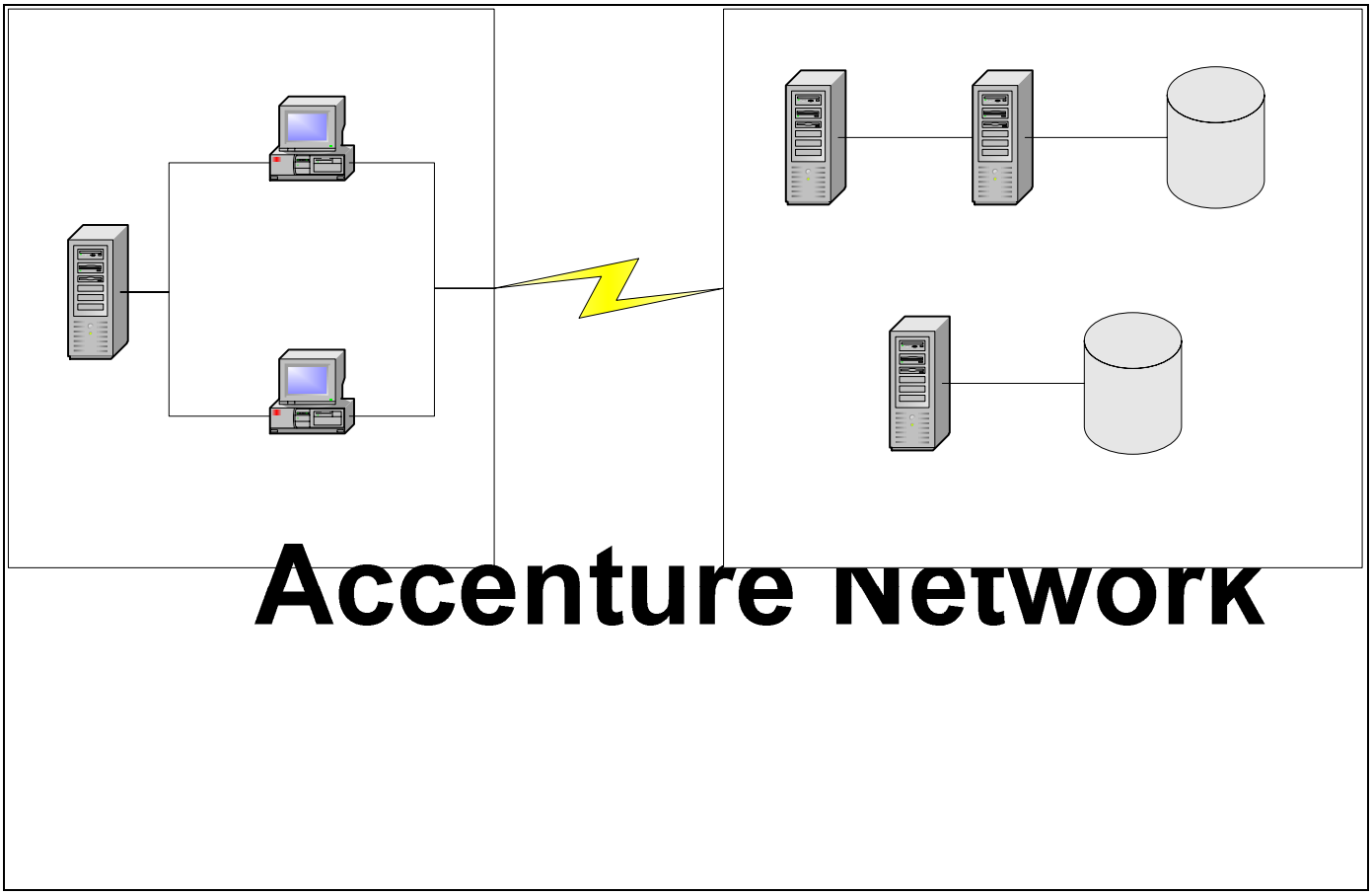
The system will be testing using an environment housed at the VDC. For system testing of the Release 1.01 (R1.01) effort, the EZ3 web application will be used. There are 4 separate web applications located at the VDC for eZ-Audit use.
EZ1: System Test of Release 1.0
EZ2: Training
EZ3: System Test for Release 1.01

3.2.2.1 Technical Architecture

The eZ-Audit test environment is composed of 4 primary components:

1. **ITA Test Environment** – Includes a web server (IBMHTTP), application server (IBM WebSphere 3.5.3), RCS components, and database instance for testing. The test servers are hosted at the VDC.
2. **Rational ClearCase** – Used for version control and source code management of application code and configuration modules.
3. **Development workstations** – Primary development machines for application and report development with connections to both the ITA Development Environment at the VDC and the Rational ClearCase code repository. Also used to migrate code to the VDC.
4. **ITA Staging Environment** – Includes a web server (IBMHTTP), application server (IBM WebSphere 3.5.3), RCS components, and database instance for testing. This environment is modeled as closely to the production environment as possible and is used for migrating changes into production.

The diagram below illustrates the test environment components.



3.2.2.1.1 Oracle Connection

SERVICE_NAME: EZTST
HOST: 4.20.15.15
PORT: 1654
EZ1: UID – systest
PWD – systest
EZ2: UID – astest
PWD – astest
EZ3: UID – uattest
PWD – uatest

Java Developer Workstation

3.2.2.2 Test Environment Migration Process

The following procedure will be used to update the test environments:

1. Download all files to be moved to the environment to your local machine.
2. Import only the Java files into your Visual Age for Java environment.
3. Export the .class files into your local directory.
4. FTP all files to the /www/dev/ezaudit/3/ directory on su35e5 (For system test of Release 1.01)
    a. All Java class files are located in the lib folder
    b. All properties and configuration files are in the servlets folder

Rational ClearCase

MicroStrategy

c. All JSP files and struts config are located in the web folder.
d. If any of the files are new, the permissions will have to be changed to 777

5. Restart WebSphere.
    e. Telnet into 4.20.25.135
    f. Login (you will need to apply for an id for this machine with ED)
    g. Execute the following command:
    sudo /opt/dev35/WebSphere/AppServer/bin/restart_ezaudit.sh
6. All images, stylesheets, and javascript files must be loaded onto su35e2 in the /www/dev/ezaudit/htdocs directory.

3.2.3 Production (Execution) Environment

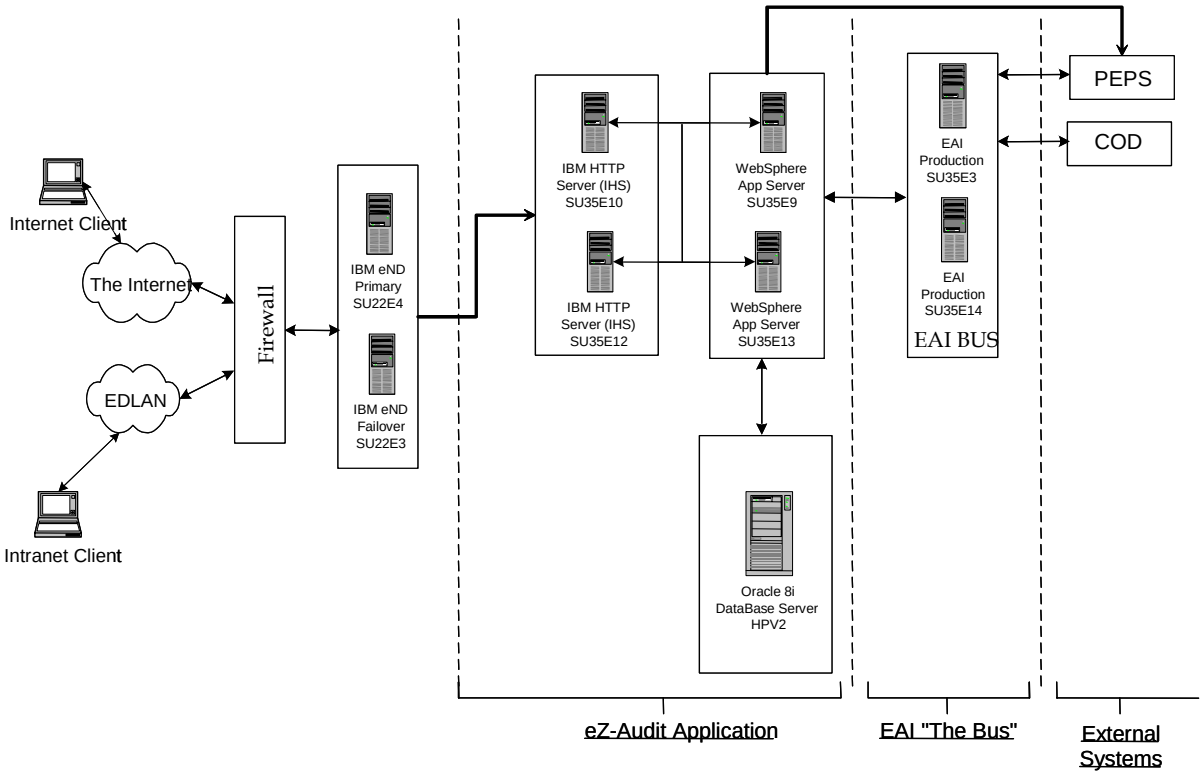
The production environment will also be housed at the VDC. All code is migrated from the test servers to the production environment using the ECM request process. Changes occur on Sundays, between 5 AM and 11 AM.

3.2.3.1 Technical Architecture

The following chart lists the primary components of the production environment:

Component	Application
Firewall	IBM Checkpoint
Load Balancer	IBM eND
Web Server	IBM HTTP Server
Application Server	IBM WebSphere Application Server Advanced Edition
Database	Oracle 8i
Enterprise Application Integration (EAI)	IBM MQSeries
Non Financial & Audit School Data	PEPS

The diagram below illustrates how the environment components work together.



3.2.3.1.1 Oracle Connection

SERVICE_NAME: EZAUDIT
HOST: 4.20.15.40
PORT: 1657
Username and Password can be obtained from the development team.

3.2.3.1.2 PEPS Integration

The PEPS integration is a series of updates that occur to ensure eZ-Audit has the most update information. There is documentation on each of these updates available, please reference them for detailed information. There are several integrations:

School File: The EAI team places the school file in a directory on su35e9 and runs the ezaudit_peps_school_file.sh script to update institution data.

Interval Data: A nightly pull from PEPS, eZ-Audit pulls submission, school group, audit firm and zone/fye data.

Clearinghouse: A file is placed in a directory that gets emailed from Sherry Quade. The development team runs a script to enter the FAC ACN's into eZ-Audit

Connection: Service_Name: PEPS.WORLD
HOST: 4.20.3.252
PORT: 1521

3.2.3.2 Production Migration Process

Moving code to the production environment requires the involvement of CSC. eZ-Audit developers do not have access to move files to the production environment. The following steps must be followed in order to move code successfully to the environment.

1. Modify and test the code in both the Assembly test and System Test environments.
2. Once the code has been verified, an ECM ticket must be used to request a change.
   A strict timetable is in place for CSC changes:
   > 10 days – Normal Change
   < 10 days – Alert Change (These changes are necessary sometimes, but if they can be avoided it is better)
   Dynamic Change (Use this for database stored functions if they do not require a restart of WebSphere. Better if a Normal Change is used)
   Emergency – An emergency code fix needs to be made.
   a. The ECM tool is a Rational ClearQuest tool. An id must be created for the user, please contact Samson Abebe if any help is needed.
   b. When logging in, go to the ED ClearQuest home page and select the VDCMP database from the dropdown list.
   c. When logged in, you may search for past changes to eZ-Audit for examples.
   d. At the top of the page, click the submit button and the drop down box next should read 'Change Request Form'.
   e. Fill in the appropriate information. Typically we will FTP files to the /www/dev/ezaudit/prod folder on su35e5. This allows consistency when moving files. Be as specific as possible in the change request and make sure to include instructions to restart WebSphere. Be sure to list any database dependencies by ECM ticket number.

f. After saving the ECM, click on the drop down box at the top and select 'Submit to DC'. The change will not go to CSC for approval unless this is selected.
g. Follow the same process and complete a separate ECM ticket for any database updates that correspond. Be sure to list any code ECM as a dependency. If there is a script that can be sent, attach the script to the ECM ticket.
h. Submit the data ECM to DC.
i. Move the files to go into production into the /www/dev/ezaudit/prod folder on su35e5.
j. Attend the weekly VDC production call the Sunday prior to the change. Contact Jack.Gilloti@csc.com to be included on the distribution list for the call.
k. Join the Sunday bridge to test the changes to production.

Microstrategy Migration:
All migrations to production must also have an ECM on file, but the eZ-Audit development team completes the changes. Two ECM requests must be included – one for the database views and one for the developer to do the work. On the Sunday of the change, the developer must confirm the database change has taken place and then begin his/her changes. More information on making changes can be found in the Microstrategy Developer's Guide.

3.2.4 Staging Environment

The staging environment was created to be as close to the production environment as possible. The process for changing the staging environment is the same as production, follow the same steps as above. The normal change window for a move to staging has not been decided, but the typical change timeframe should be as follows:

1. SIR is submitted by client
2. SIR is logged, fixed, and tested.
3. 4 ECM tickets are logged. One for data and one for code for the next Sunday that is > 10 days out. One for data and one for code that is the Thursday before the Sunday move.
4. Once the code is moved into staging, test the fix to make sure there are no issues
5. Then test the changes in production on Sunday without changing any of the data.

3.2.4.1.1 Oracle Connection

Service Name: EZAUSTG
HOST: 4.20.15.15
PORT: 1693

3.3 Configuration Management Roles and Responsibilities

The following section contains information on roles and responsibilities associated with implementing CM at eZ-Audit. Reference the Project Plan for a comprehensive description of eZ-Audit roles and responsibilities. The roles and responsibilities listed here are specific to CM activities.

3.3.1 Kelly Tate– Project Partner/Engagement Partner

- Use CM Reports to gain visibility into project
- Resolve escalated CM issues and act on CM metrics, as appropriate
- Ensure team members are knowledgeable of CM concepts and techniques and that they are applied to project activities
- Prepare and conduct performance evaluations of project managers with respect to CM responsibilities

3.3.2 Ti Baker – FSA Project Manager

- Review and approve the CM Plan
- Review all CM Reports
- Follow the CM processes and procedures outlined in this plan
- Establish the overall project schedule for CM activities with CM Manager
- Ensure the definition and maintenance of all development processes and standards, including CM
- Participate as a CM Deliverable Owner for eZ-Audit as appropriate
- Participate as a CM-related SQA Reviewer for other projects as appropriate
- Verify that all audits scheduled to be conducted as outlined in this plan are conducted

3.3.3 Rondell Milton – PI Liaison or Designated CM Reviewer

- Ensure eZ-Audit compliance with the FSA's SLC CM standards and procedures set by the CM Manager, the CCB, and the QPI team.
- Ensure that periodic baseline audits occur at eZ-Audit
- Follow CM audit processes and procedures as outlined in this plan when conducting audits of eZ-Audit CM activities

3.3.4 Sherry Quade/Cricket Hosier – System Integrity/Data Owner

- Ensure interface with PEPS and external entities are accurate (e.g. institution and clearinghouse updates)
- First point of contact for FSA eZ-Audit help desk

3.3.5 Barbara Johnson – System Security Officer

- Ensure eZ-Audit compliance with FSA security standards
- Owner of all security documentation, policies and procedures

3.3.6 System Users

- Use eZ-Audit for online submission and remediation of financial statements and compliance audits.

3.3.7 Andre Sakaluk - Configuration Management Manager

Andre Sakaluk will act as the project's Configuration Management Manager. eZ-Audit CM Manager will prepare the CM Plan with assistance from the Project Manager. The CM Manager is responsible for updating the plan according to the processes set forth in the plan and communicating those updates to the project team.

The CM Manager has the following responsibilities:

Plan CM

- Identify configuration items (CIs) to be managed by the CM function
- Appoint members to the Change Control Board
- Create, manage, maintain, and communicate the CM Plan and any CM standards and procedures to all stakeholders
- Ensure that all project team members involved in CM receive training on their roles and in how to perform their activities and how use CM tools, such as the e-project file repository
- Make updates to CM Plan, as appropriate, and only after approval by the Change Control Board
- Ensure that any updates to the CM Plan are communicated to appropriate project team members
- Establish project schedule for CM activities with Project Manager
- Form and manage a CM team
- Conduct performance review for members of the CM team

Implement Changes

- Create products from the e-project file repository as authorized by the Change Control Board
- Process and track change requests, and subsequent updates to the configuration management library
- Perform regression tests on all trial baselines to ensure that implemented changes have not caused problems

- Coordinate reviews of configuration requests (CRs) with the Change Control Board

Track and Report on CM Status/Audits

- Ensure that the integrity of all CIs is maintained by maintaining status of all CIs and tracking problem reports associated with them
- Conduct audits of CM activities as planned, including performing a baseline audit prior to closing baselines
- Track, report, and communicate CM status and audit reports to the Project Manager and Project Partner
- Ensure that configuration item change requests and problem reports for all configuration items are initiated, recorded, reviewed, approved, and tracked according to the procedure documented in this plan

Maintain Library

- Create and manage, including security, the e-project file repository
- Ensure that project team members have the appropriate access to the configuration management library

3.3.8 Change Control Board

The members of the Change Control Board include:
- Ti Baker, Change Control Board Head/Chairperson
- FSA Core Team, Change Control Group Members
- Brian Cannavan, Technical Project Lead
- Seth Sinclair, Functional Project Lead

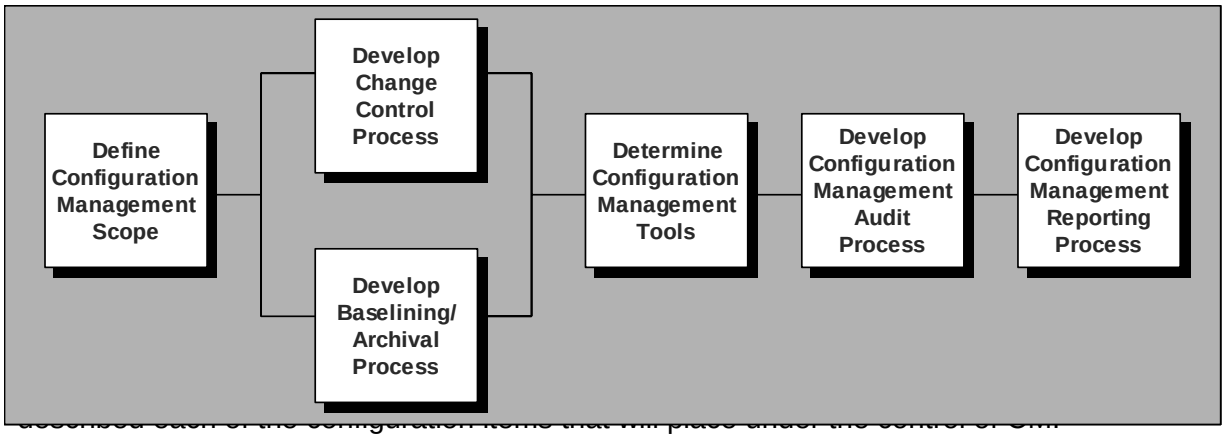
Although the CM Manager has overall responsibility for CM at eZ-Audit, a Change Control Board (CCB) will be established to evaluate and authorize the changes to the system components that have been identified as configuration items. The CCB has authority to manage the project's baselines. The CCB has the following responsibilities:
- Authorize the establishment of baselines and the identification of configuration items
- Monitor changes and updates to project requirements as part of CM
- Represent the interests of the project manager and FSA
- Review and authorization of changes to the baselines
- Authorize the creation of products from the configuration management library
- Establish, document, and communicate criteria by which Configuration Requests (CRs) will be evaluated
- Review the CM plan
- Review and approve all new baselines
- Create and communicate minutes CCB meeting minutes to affected groups

The impact of CCB change requests will be handled by the project. An assessment of the project impact will determine how the change request will affect the nature of the project.

The CCB will meet bi-weekly starting in May 2002.

3.4 Configuration Management Process Flow

eZ-Audit Configuration Management process flow contains six steps:



described each of the configuration items that will place under the control of CM.

3.4.2 Develop Change Control Process

During the Develop Change Control Process step, eZ-Audit developed a process for modifying the configuration items that will place in CM. This change control process outlines the process used to request, evaluate, approve or disapprove, and implement changes to baselined configuration items. This process ensures that changes to the configuration items are authorized and are performed in an orderly and appropriate sequence.

3.4.3 Develop Baselining/Archival Process

During the Develop Baselining/Archival step, eZ-Audit developed the process that will use to store and maintain all of the identified configuration items.

3.4.4 Determine Configuration Management Tools

During the Determine the Configuration Management Tools step, eZ-Audit identified and implemented the tools necessary for CM activities. Further, eZ-Audit will outline the process for using these tools.

3.4.5 Develop Configuration Management Audit Process

During the Develop the Configuration Management Audit Process step, eZ-Audit identified the process for conducting audits of eZ-Audit CM activities. At a minimum, audits will be performed between project phases. Therefore, an audit will be conducted before the project proceeds to design, build, assembly/integration testing, system testing, acceptance testing and production.

3.4.6 Develop Configuration Management Reporting Process

During the Develop Configuration Management Reporting Process step, eZ-Audit identified the process that will be used to report on eZ-Audit CM activities.

3.5 Configuration Management Scope

The CM Manager and the CCB established the items that will be place under the control of Configuration Management. These items are known as configuration items (CIs) (Please see section 3.5.2. for the complete list). Items include deliverables identified in the Technical Proposal along with other Project Management documents.

3.5.1 Configuration Item Naming Conventions

The eZ-Audit will utilize the following standard naming conventions for all CIs:

The standard naming convention for files in the eZ-Audit e-project file repository includes the following:

Description_date_version#

The description of the file should convey (even to the uninitiated) the basic idea about what a given file is. For example, a file containing the project workplan should be called:

"eZ-Audit Workplan_08082002_V3"

as opposed to:

"eZ-Audit WP_08082002_V3".

The format of the date as represented in the file's name should be in the following format: _mmddyyyy

Versions of files should be represented in the following format: _V#

3.5.2 Configuration Items

eZ-Audit has identified the following configuration items (CI) to be placed under CM. A CI is defined as a work product that will require configuration control. A CI may be a single piece of work or a group of files that together form the basis for a single program or document. The CIs identified are grouped together in numerous baselines based on project needs (e.g., requirements baselines, specifications baselines, design baselines, code baselines, integration baselines (system and component), operational baselines). All change that occurs to the baselines during the project are being accounted for and tracked.

CI Name	Type	Description	Baseline Name*	Date or Phase CI Placed under CM	Owner	Repository / Path	Level of CM
Business Case	Project Management	Deliverable	EFS Business Case	Vision Phase	Gene Murphy	Modpartner.eproject.com/TO86/Documents	Version control on e-project

CI Name	Type	Description	Baseline Name*	Date or Phase CI Placed under CM	Owner	Repository / Path	Level of CM
						/Project Management/ Business Case and Technical Proposal	
Statement of Objectives	Project Management	Deliverable	N/A (Statement of Objectives included in Project Plan and Task Order)	Vision Phase	Gene Murphy	N/A (Statement of Objectives included in Project Plan and Task Order)	Version control on e-project
Task Order	Project Management	Deliverable	TO86 EAFS (Mod 01) Tech Prop 3-19-02.doc	Vision Phase	Gene Murphy	Modpartner.eproject.com/TO86/Documents/Project Management/ Task Order	Version control on e-project
Project Work Plan	Project Management	Deliverable	eZ-Audit Work Plan	Vision Phase	Gene Murphy	Modpartner.eproject.com/TO86/Documents/Project Management/ Project Work Plan	Version control on e-project
Project Plan	Project Management	Deliverable		Vision Phase	Gene Murphy	Modpartner.eproject.com/TO86/Documents/Project Management/ Project Plan	Version control on e-project
Configuration Management Plan	Project Management	Deliverable	eZ-Audits Configuration Management Plan 0117 V1.doc	Vision Phase	TBD	Modpartner.eproject.com/TO86/Documents/CMMi/CM	Version control on e-project
Requirements Document	Software requirements	Deliverable	86.1.2. eZ-Audit Requirements Deliverable	Definition Phase	Carrie Marks	Modpartner.eproject.com/TO86/Documents/Deliverables/ 86.1.2. eZ-Audit Requirements	Version control on e-project
System Requirements Review (SRR) Results Document	Software requirements	Deliverable	Requirements Deliverable Feedback Change Log_03262002_V1	Definition Phase	Carrie Marks	Modpartner.eproject.com/TO86/Documents/Deliverables/ Requirements Deliverable Feedback Change Log_03262002_V1	Version control on e-project
Preliminary Design Document	Software design	Deliverable	eZ-Audit Preliminary Functional Design Document	Definition Phase	Matt Williamson	Modpartner.eproject.com/TO86/Documents/Deliverables/ eZ-Audit Preliminary Functional Design Document	Version control on e-project

CI Name	Type	Description	Baseline Name*	Date or Phase CI Placed under CM	Owner	Repository / Path	Level of CM
Preliminary Design Review (PDR) Results Document	Software design	Deliverable	N/A (we received no comments on this deliverable)	Definition Phase	Matt Williamson	N/A (we received no comments on this deliverable)	Version control on e-project
Status Reports	Project Management	Weekly status updates	Status Report	Vision Phase	Team	Modpartner.eproject.com/TO86/Documents/Administration/Status Reports	Version control on e-project
Metrics Workbook	Project Management	Monthly metrics tool	eZ-Audit Metrics Workbook for March 2002	Vision Phase	CMMi Lead	Modpartner.eproject.com/TO86/Documents/Metrics Workbook	Current month versions are maintained; prior versions read-only
Status Meeting Minutes	Project Management	Meeting notes	e-Audit Bi-Weekly Status 02-01-22	Vision Phase	Team	Modpartner.eproject.com/TO86/Documents/Administration/Meeting Minutes	Read only on e-project upon completion
SQA Reports	Software requirements	Software Quality Assurance Reports	sqa_report_template_03192002_V1	Vision Phase	IV&V	Modpartner.eproject.com/TO86/Documents/CMMi/SQA Reports	Version control on e-project
CM Audit Schedule	Project Management	CM Audit Schedule	CM Audit Schedule 03192002 V1	Vision Phase	Seth Sinclair	Modpartner.eproject.com/TO86/Documents/CMMi/CM Audit Schedule	Version control on e-project
CM Audit Reports	Project Management	CM Audit Reports	CM Audit Report Template 03192002 V1	Vision Phase	Seth Sinclair	Modpartner.eproject.com/TO86/Documents/CMMi/CM Audit Report	Version control on e-project
Change Requests	Software requirements	Change Requests	Requirements Change Request Form	Vision Phase	Team	Modpartner.eproject.com/TO86/Documents/Change Requests	Version control on e-project
Templates	Project Management	Templates	N/A	Vision Phase	CMMi Lead	Modpartner.eproject.com/TO86/Documents/Templates	Version control on e-project
Deliverable Tracking	Project Management	Deliverable Tracking	N/A	Vision Phase	Team	Modpartner.eproject.com/TO86/Documents/Deliverables	Version control on e-project
Risk Matrix	Project Management	Risk Matrix	N/A (Risk Matrix located in the bi-weekly status report)	Vision Phase	Gene Murphy	N/A (Risk Matrix located in the bi-weekly status report)	Version control on e-project
Issues Log	Project Management	Issues Log	N/A	Vision Phase	Gene Murphy	Modpartner.eproject.com/TO86/Documents	Version control on e-project

CI Name	Type	Description	Baseline Name*	Date or Phase CI Placed under CM	Owner	Repository / Path	Level of CM
						/Issues	
E-Audit Contact List	Project Management	E-Audit Contact List		Vision Phase	CMMi Lead	Modpartner.eproject.com/TO86/Administration/Contact List	Version control on e-project
Requirements Traceability Matrix	Software requirements	Requirements Traceability Matrix	Matrix is located in Rational	Vision Phase	Matt Williamson	Matrix is located in Rational	Version control in Rational
Use Cases	Software design	Use Cases	86.1.4 eZ-Audit Functional Design.zip	Design Phase	Matt Williamson	Modpartner.eproject.com/TO86/Deliverables/86.1.4 eZ-Audit Functional Design/Use Cases	Version control on e-project

*A baseline is a set of configuration items that has been formally reviewed and agreed upon, which then serves as the basis for further development, and can be changed only through the formal change control process. Usually similar configuration items are grouped in the same baseline, such as all of the requirements being a part of the Requirements baseline.

3.6 Change Control Process

During the development and maintenance processes, controlled baselines will change and configuration control is designed to manage the changes. Changes may be needed for a variety of reasons, such as add new technology or functionality, correct problems, and respond to technical and operational tests and evaluations. The overall change process for the eZ-Audit project is illustrated in the following diagram:

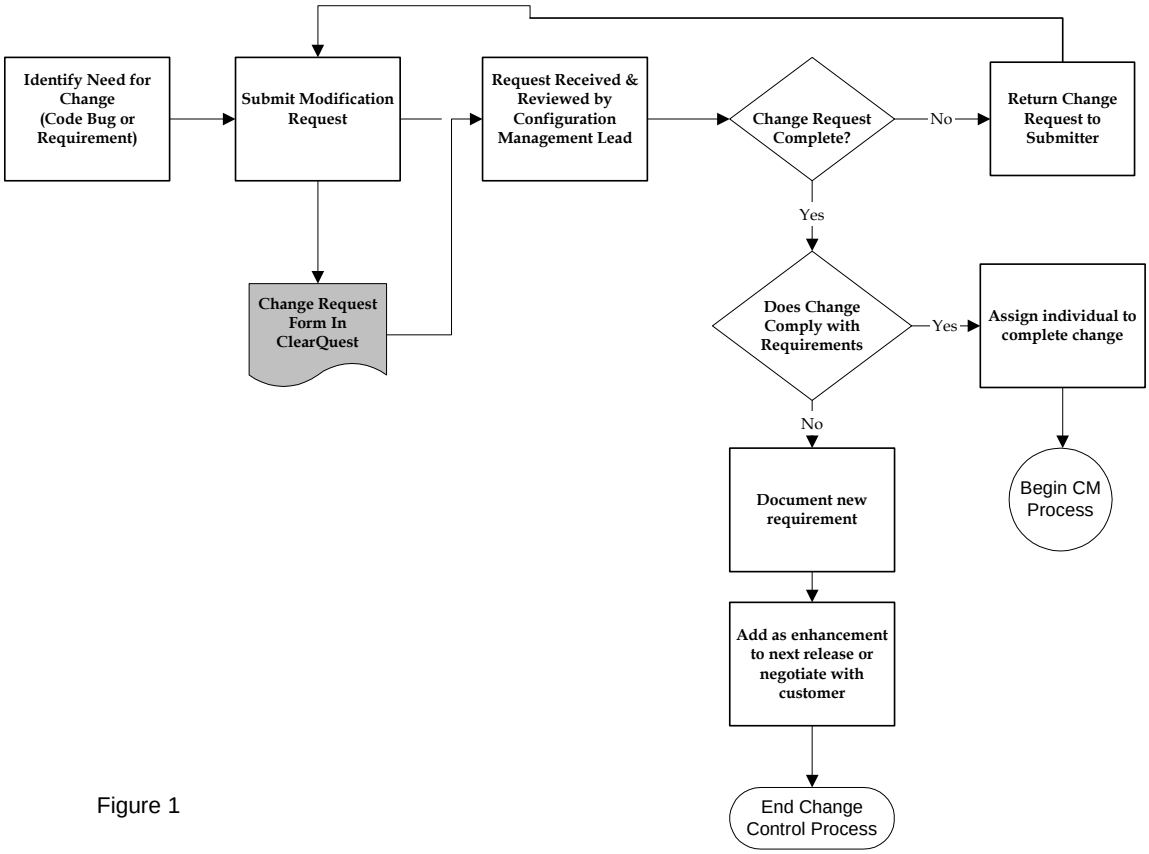


Figure 1

3.6.1 Vision and Definition Documentation

After CIs are placed under CM, a Configuration Request (CR) will need to be completed and authorized by the CCB for the CI to be modified. The process by which CIs are modified is known as the Change Control Process. By following these processes, eZ-Audit will ensure that only approved changes are incorporated into the CIs, thus ensuring their integrity.

In general, the change control process consists of four steps:

1. Request for Change – A CR should be completed and submitted to the CCB to modify a CI, or group of CIs. A CR may result from desired additions, modifications, or deletions in capabilities of the CI. CRs may also result from problems detected during verification and validation.
2. Evaluation of Change – All CRs must be reviewed and approved by the CCB. CRs will be either approved or disapproved. Those CRs that are disapproved will be updated with comments explaining why the request was not approved. The CRs that are disapproved do not move on to step 3.
3. Implementation of Change – All approved CRs must be implemented according the procedures set forth in this plan. The implementation of the CRs approved by the CCB

includes version control and subsequent baselining of all CIs. This includes recording, maintaining, and providing details of all changes to CI.

4. Notification of Change – Groups affected by the CRs will be notified of changes through the distribution of the minutes from each CCB meeting.

eZ-Audit will implement the following CR processes to ensure the integrity of its CIs.

3.6.1.1 CR Creation

CRs will be created using the CR tool, as described in section 3.8. Designers, developers, system testers and managers can create CRs. A CR should be created when a project team member identifies a necessary change for a CI. A CR can be submitted to change a single CI, or a group of related CIs.

3.6.1.2 CR Evaluation

CRs will be evaluated at each bi-weekly CCB meeting. All CRs will be evaluated based on criteria that has been established by the CCB and approved by the Project Partner, Project Manager, and CM Manager. The following criteria will be used to evaluate eZ-Audit CRs:

- Size of change
- Complexity of change in reference to related system
- Date when change is needed
- Impact of the change on current and subsequent work
- Change implementation costs
- Change implementation benefits
- Critically of area involved when making a change
- Approved changes already in process
- Test requirements of change
- Security Implications

CRs that are approved by the CCB will be implemented based on the procedures set forth in this plan. Those CRs that are not approved will be updated with comments explaining why the change was not approved.

3.6.1.3 CR Implementation

Changes to CIs will be implemented using the Configuration Management Library tool, as described in section 3.8. Only CRs that have been approved by the CCB will be implemented using the Configuration Management Library tool. To implement a CR on a CI, the following check in/check out process will be followed.

The check in/check out procedure is as follows:
The user locates the document on the e-project tool.
The user checks out the document.
The user downloads the document to a local drive.
After making the changes, the user uploads the new version to the e-project including version notes and explanation of the change.
The user checks in the document.

A current status and history will be maintained for each CI through the use of a change log or CM Library tool. The Configuration Management Library tool will ensure that configuration management activities will be recorded in such a manner that the content and status of each CI is known and that previous versions can be recovered. eZ-Audit has developed a numbering convention that will be used to track versions of the CIs so that each version of the CI can be uniquely identified and retrieved if necessary.

3.6.1.4 CR Notification

Changes to CIs resulting from approved CRs will be communicated to affected groups through CCB meeting minutes.

3.6.2 Requirements Management

E-Audit requirements management process will follow the same configuration management procedures described in the section 3.6.1.

Prior to the incorporation of requirements into future deliverables, project management will review all requirements to ensure that:

- Incomplete or missing allocated requirements are identified,
- Allocated requirements are reviewed to ensure that they are feasible and appropriate to implement in software, clearly and properly stated, consistent with other requirements, and testable, and
- Allocated requirements that have been identified as having problems are reviewed and analyzed with the appropriate groups and necessary updates are made.

Project management will ensure that the project team understands how all allocated requirements will be implemented and that the project team understands that it is responsible for incorporating the requirements into the appropriate deliverables.

eZ-Audit will use the Requirements Traceability Matrix to track requirements through all phases of the project. The Requirements Traceability Matrix will demonstrate how eZ-Audit requirements

have been implemented by referencing requirements to design documents, module code, and test scripts that result from them.

The eZ-Audit project team will use several Rational tools and the Rational Unified Process, known as RUP, to manage all configuration changes to the requirements. The eZ-Audit Team has imported all requirements into Rational RequisitePro and begun using RequisitePro to manage all requirements for the eZ-Audit application. To ensure all requirements have been met, a Requirements Traceability matrix has been created in RequisitePro to trace each requirement to one or multiple Use Cases.

A Requirements Traceability matrix set up in Rational identifies the functional and technical requirements for the eZ-Audit application, as compiled and recommended by the eZ-Audit Subject Matter Team. eZ-Audit will control updates to the Requirements Traceability Matrix and Requirements Baseline by the same configuration management procedures described in the section 3.4.

3.7 Baselining Process

Baselining is the core of configuration management. By baselining, eZ-Audit can control and measure the change that occurs to its CIs in a given period of time and ensure the integrity of the product that is released. The CCB authorizes the creation of products from the e-project file repository tool. The products of eZ-Audit, for use both internally and externally, will be released from the e-project file repository tool only. Further, products that are released from the e-project file repository tool will be built only from CIs within the tool. Once a set of products is released from the baseline, the baseline is archived and a new baseline is taken.

eZ-Audit will support the following baselines:
- Requirements documentation baseline
- Design baseline
- Internal Document baseline
- Project Management baseline

Baselines will be taken upon mutual agreement on the document content. Baselines will also be taken at the end of the first phase of work (Vision and Definition).

- Production Code Release 1.0 Baseline Freeze Date – 4/1/2003
- Requirements baseline freeze date for eZ-Audit Release 1.01 – 5/6/2003
- Production Code Release 1.01 Baseline Freeze Date – 7/31/2003

3.8 Historical Archival Process

After a baseline has been approved and released, the baseline will be archived on e-project. The project's historical archival process will retain information about each revision of the CIs, thereby, allowing the team access to previous versions of a particular baseline, if necessary.

3.9 Migration Processes

The migration process varies for each environment. Please see section 3.2 for a detailed explanation of the environments.

1. Development Environment – code is migrated to server via FTP by the developers
2. Test Environment – code is migrated from Development environment to the Test Environment via FTP. Also controlled by developers.
3. Staging Environment – code is migrated from testing server to the staging server. Process is controlled by request to VDC and migrated by VDC personnel.
4. Production Environment – code is migrated from testing server to the production servers (su35e9 and su35e13). Process is controlled by request to VDC and migrated by VDC personnel.

3.10 Configuration Management Tools

The configuration management tools are housed within the VDC. All procedures and policies regarding these tools are the responsibility of CSC and are governed by the Service Level Agreement (SLA) in place. Detailed information can be obtained in the FSA Memorandum of Agreement and Service Level Agreement with the VDC (CSC).

The Change Control Board Spreadsheets have following content:

- CR Number
- Status of CR
- Date opened
- Date closed (planed and actual)
- Priority of change
- Name and E-mail of CR initiator
- Description of change
- Name of CI(s) to be changed
- Impact of change on current and subsequent work
- Related CRs
- Costs/Benefits of change
- Test requirements of change
- Resources needed for making the change

The eProject file repository has abilities to:

- Support multiple control levels of CM
- Provide for the storage and retrieval of CIs
- Provide locks to prevent unauthorized changes
- Provide sharing and transfer of CIs between affected groups and between control levels in the library
- Help in the use of product standards for CIs
- Provide for the storage and recovery of archive versions of CIs
- Help to ensure the correct creation of products from the CM library

- Provide for the storage, update, and retrieval of CM records
- Support production of CM reports
- Provide for the maintenance of the library structure and contents

3.10.1 Using the CR tool

Using of the Change Control Board Spreadsheets consists of:
- Locating the spreadsheet
- Completing the requested fields
- Presenting the CR in the CCB meeting
- Updating the resolution
- Assigning/making the change
- Communicating the change

3.10.2 Using the CM Library Tool

Using of the e-project file repository tool consists of:

- Locating the document on the e-project tool
- Checking out the document
- Downloading the document to a local drive
- After making the changes, uploading the new version to the e-project including version notes and explanation of the change
- Checking in the document

3.10.3 Using the Issue Tracking Tool for CM

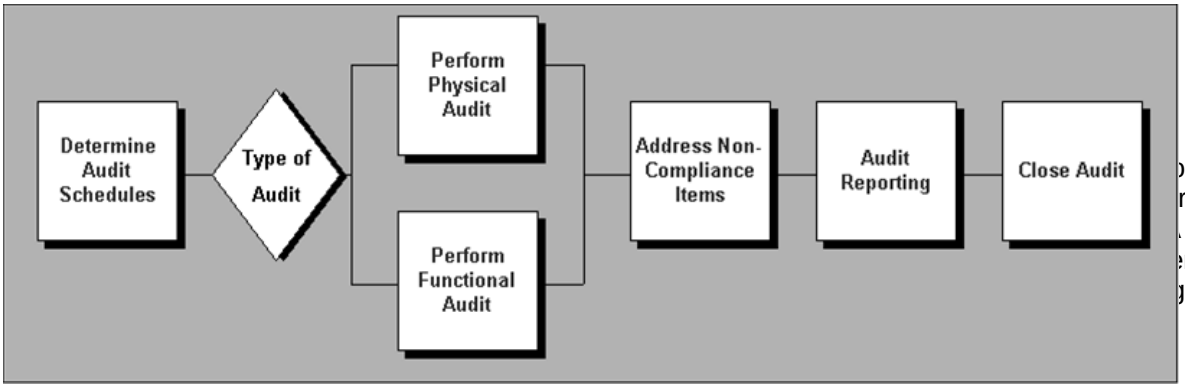
E-project file repository contains the Issues Log tool. The following Items can be logged for each issue:
- Project
- Topic
- Description
- Start Date
- Deadline
- Priority
- Budget Impact
- Percent Complete

3.11 Disaster Recovery Procedures

This section will be updated in the appropriate phase of the project's lifecycle.

3.12 Configuration Management Audit Process

Several audits will be performed on the CM activities of eZ-Audit, including an audit of the baselines in the e-project file repository, to ensure that project team members follow the processes and procedures in this plan. The following process flow will be used to complete CM audits:



ocess.
rmed
. new
ensure
gement

The auditor will perform two types of audits as outlined in the CM audit process flow: a Functional Configuration Audit (FCA) or a Physical Configuration Audit (PCA). A FCA ensures the integrity of the requirements by cross-referencing audit items to the requirements from which they were generated. A PCA ensures the integrity of the Configuration Management baselines, configuration items and tools used to implement CM.

When auditing the e-project file repository tool, the auditor will:
- assess the integrity of the baseline,
- review the structure and facilities of the e-project file repository tool,
- verify the correctness and completeness of the baseline contents,
- verify that changes to the baseline have been implemented as intended, and
- verify compliance with CM processes and procedures.

The results of each audit will be reported to the Project Manager and other necessary parties. The CM Manager will track all action items until closure and report on the status of the action items to the Project Manager and to the project's PI Liaison.

Beyond the CM audits that will be performed by the CM Manager and the independent assurance authority, the eZ-Audit PI liaison/CM reviewer designated by the PI liaison is responsible for reviewing the project's CM activities and work products and reporting the results using the Best Practices Matrix tool located at location.

The PI liaison/CM reviewer designated by the PI liaison will conduct two reviews/audits during each project phase. When conducting the audits, the PI liaison/CM reviewer designated by the PI liaison will verify:
- Compliance with the CM standards and procedures by the CM Manager and team, the CCB, the Quality and Process Improvement Team (QPI) group, and other affected groups.
- Occurrence of periodic baseline audits.

Further, when the PI liaison/CM reviewer designated by the PI liaison conducts an audit, the processes outlined for CM audits will be followed. The PI liaison/CM reviewer designated by the

PI liaison will report the results of the review/audit using the Configuration Management Audit Template. The report will be distributed to project management for review.

3.13 Configuration Management Status Reporting

In order to give eZ-Audit Project Management high-level insight into the overall status of CM activities, CM status will be reported in the CM Status Report. Project management is responsible for reviewing the report and taking necessary action to resolve potential issues. If necessary, CM issues will be escalated and addressed by the Engagement Partner.

The CM Status Report will be delivered weekly and will contain the following information:
- Technical, cost, staffing and schedule performance as compared to the CM plan
- Resource use as compared to estimate
- CM conflicts or issues
- Dependency between groups
- CM risks
- Action items. Action items should appear on the reports until they are closed

As part of the CM Status Report, a Configuration Item Status Report (CISR) will be prepared periodically for the CM Manager and other affected groups. The CISR provides an in-depth status accounting of all of the project's CIs.

3.13.1 Configuration Item Status Reporting

eZ-Audit will utilize the Configuration Item Status Report (CISR) section of the Configuration Management Status Report to maintain a continuous record of the status of all CIs. The CISR provides an in-depth status accounting of all the project's CIs and is the record keeping function of CM.

The CISR will track the following information against the CM Plan and CM baselines:
- Time at which each baseline is established
- When each CI was included in the baseline
- A description of each CI
- Status of each CI related change
- Description of the change to the CI
- Documentation status of each baseline
- Changes planned for each identified future baseline

The CISR will be compiled weekly and distributed to the CM Manager and other affected groups with the CM status report. If necessary, issues identified in the CISR will be escalated and resolved by the Project Partner and/or Project Management.

By performing this status accounting of CIs, the CM Manager will be able to determine how the CM program is progressing in relation to documentation required, baselines established, changes and the rate at which changes are occurring. A review can be made of the type of changes, reasons for changes and cost of changes scheduled.

4.0 Configuration Management Training

Refer to the project's Training Needs Matrix located at location for comprehensive listing of CM related training. Team members that are affected by CM, including the QPI group, will be trained in CM processes and procedures and how to use the configuration management tools: Change Information Change Request and Configuration Management Library.

5.0 Configuration Management Schedule and Cost

The CM schedule and milestones for eZ-Audit are identified in the table below. Identified activities are eZ-Audit' internal audit and checking mechanism.

Activity/Task Name	Start Date	End Date	Owner	# of Resources Required	Estimated Budget	Notes
Post Vision Audit	02/04/2002	02/04/2002	Maja Dragnic	1	6 hrs	
Post Definition Audit	05/16/2002	05/16/2002	Maja Dragnic	1	6 hrs	

6.0 Interface Control

This section will be updated in the appropriate phase of the project's lifecycle.

7.0 Configuration Management Metrics Tracking

7.1 Objective

By using metrics, Accenture can consistently demonstrate improvement. Improvement is reflected in measures of the projects' effectiveness, productivity, and quality of product and processes. In order to continuously improve, a process must be regularly "measured" and modified.

7.2 Process

The eZ-Audit PI Liaison will work with the CM Manager to collect the metrics listed below. PI Liaison will be responsible for analyzing the metrics data and communicating the results to the PMO office (when applicable) and to project management through the CM Status Report.

eZ-Audit
Configuration Management Metrics

The configuration management metrics tracking process will track the following metrics and create a monthly report:

- Total number of changes for all environments
- How many changes were disapproved
- How many changes are currently open
- How many have been opened since the last report
- How many have been closed since the last report

8.0 Appendices

8.1 Change Request Form

Use the following form to request changes to the eZ-Audit Application. Fill in all fields and send the completed form to seth.b.sinclair@accenture.com. Please use a separate form for each individual Change Request. All Change Requests will be reviewed by the CCB during project status meetings meetings. Status of each proposed Change Request will be documented in the CCB Tracking tool.

Title:

Author:

Date Submitted:

Requirement:

Requirement Change Request Description:

Security Impacts:

Priority (indicate with an "X"):

High	(Essential for core capability deployment; no identified work-around exists)
Medium	(Significant to capability deployment; work around identified, if not implemented in the system)
Low	(Desired feature, but no impact to system functionality)

==========

Enhancement Request #

Discussion Date:

Discussion (and/or options):

Decision:

Documentation Updates Required:

CCB Chairperson Signature: _____

CCB Lead Signature: _____